



Cours d'intégration

3W Academy

CSS 4. La décoration

4. La décoration

4.1. Les couleurs

4.1.1. Les noms prédéfinis

4.1.2. Le code hexadécimal

4.1.3. Le code rgb ou rgba

4.1.4. Le code hsl ou hsla

4.2. Modifier la couleur du texte

4.3. Modifier la couleur de fond

4.4. Ajouter une image de fond

4.5. Les gradients

4.6. Ajouter des arrondis

4.7. Ajouter une ombre

4. La décoration

4.1. Les couleurs

On peut déclarer une couleur de différentes façons:

4.1.1. Les noms prédéfinis

Par exemple: red.

La liste complète est disponible ici:

https://developer.mozilla.org/fr/docs/Web/CSS/Type_color

4.1.2. Le code hexadécimal

Par exemple: #FF0000.

Dans un code couleur hexadécimal, chaque composante de la couleur, rouge, vert et bleu est représenté par un octet sous la forme d'un nombre hexadécimal #FF0000. En effet on peut représenter un octet (256 possibilités) avec deux caractères hexadécimaux (0 à 9 puis A, B, C, D, E F soit 16 possibilités): $16 * 16 = 256$.

4.1.3. Le code rgb ou rgba

Par exemple: rgb(255,0,0) ou rgba(255,0,0,1).

Dans un code couleur rgb, chaque composante de la couleur, rouge, vert et bleu est représenté par un octet sous la forme d'un nombre décimal rgb(255,0,0). Il existe aussi une version avec la couche alpha qui représente la transparence. La valeur va de 0 (totalement transparent) à 1 (totalement opaque).

4.1.4. Le code hsl ou hsla

Par exemple: #FF0000.

Par exemple: `hsl(0,100%,50%0)` ou `hsla(0,100%,50%,1)`.

Dans un code couleur hsl, les composantes de la couleur sont différentes: "h" représente la teinte (hue), "s" représente la saturation et "l" la luminosité. La teinte est une valeur comprise entre 0 et 359 alors que la saturation et la luminosité sont en pourcentage. Il existe aussi une version avec la couche alpha.

4.2. Modifier la couleur du texte

Pour changer la couleur du texte, on utilise la propriété *color*.

Dans le CSS

```
.example {  
    color: red;  
}
```

4.3. Modifier la couleur de fond

Pour changer la couleur du fond, on utilise la propriété *background-color*.

Dans le CSS

```
.example {  
    background-color: rgba(0,0,0,0.2);  
}
```

4.4. Ajouter une image de fond

Pour afficher une image de fond on utilise la propriété *background-image*:

Dans le CSS

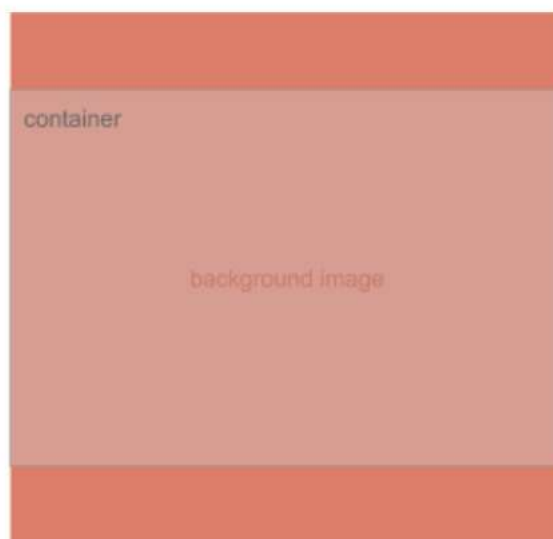
```
.example {  
    background-image: url("images/bg.png");  
}
```

D'autres propriétés permettent d'agir sur l'image de fond.

Propriété	Valeurs possibles	Description / exemple
<i>background-position</i>	Une combinaison des valeurs <i>top</i> , <i>bottom</i> , <i>center left</i> , <i>right</i> ou de valeurs numérique en pourcentage	Permet de spécifier la position de l'image de fond par rapport à son conteneur <i>background-position: right center;</i>
<i>background-repeat</i>	<u>repeat</u> <i>repeat-x</i> <i>repeat-y</i> <i>no-repeat</i>	Permet de définir la répétition de l'image de fond. <i>background-repeat: none;</i>
<i>background-size</i>	<u>auto</u> <i>contain</i> <i>cover</i> (valeur en pourcentage)	Permet de spécifier la taille de l'image de fond, soit en pourcentage par rapport à son conteneur. <i>background-size: cover;</i>

La valeur par défaut est soulignée.

Comportement attendu avec les valeurs *cover* et *contain* de la propriété *background-size*:



4.5. Les gradients

Pour ajouter un gradient de couleurs comme fond d'un élément, il faut utiliser l'une des fonctions de création de gradient comme valeur de la propriété *background-image*.

Il existe plusieurs fonctions permettant de générer un gradient de couleur

- *linear-gradient()*
- *radial-gradient()*
- *repeating-linear-gradient()*
- *repeating-radial-gradient()*

La fonction *linear-gradient()* accepte au minimum trois paramètres:

- Le premier paramètre est l'orientation: on peut utiliser des constantes (top, left, right, bottom) ou une valeur en degré, par exemple, 30deg.
- Les paramètres suivant sont les différents points d'ancrage des couleurs successives. Il y en a deux au minimum.

Dans le CSS

```
.example {
  background-image: linear-gradient(45deg, red, blue);
}
```

Dans ce premier exemple, on a un dégradé linéaire du rouge au bleu qui est orienté du coin inférieur gauche vers le coin supérieur droit.



Pour un dégradé radial, on aura par exemple:

Dans le CSS

```
.example {
  background-image: radial-gradient(circle, red, blue);
}
```



Il est possible de fixer la position de chaque point d'ancrage avec des pourcentages

Dans le CSS

```
.example {  
    background-image: linear-gradient(45deg, red 10%, blue 20%);  
}
```



On peut aussi ajouter des couleur intermédiaires. Voici un exemple plus complexe:

Dans le CSS

```
.example {  
    background-image: linear-gradient(0deg, #ff9900 0%, #cc4125 60%,  
    #674ea7 100%);  
}
```

L'enchaînement des paramètres peut être schématisé comme ceci:



Il est possible de superposer plusieurs dégradés. Il faut séparer les différentes valeurs par des virgules.

Il ne pas hésiter à utiliser des outils en ligne pour générer le code CSS qui peut être parfois complexe. Pour cela recherchez un outil avec les mots clé: **css gradient generator**

4.6. Ajouter des arrondis

Il est possible de modifier l'arrondi des boîtes avec la propriété *border-radius*.

Dans le CSS

```
.example {  
    border-radius: 1em;  
}
```

On peut spécifier chaque coin indépendamment:

Dans le CSS

```
.example {  
    border-radius: 1em 10% 0 5px;  
}
```

Avec l'ordre respectif suivant:

- haut gauche
- haut droit
- bas droit
- bas gauche

4.7. Ajouter une ombre

Pour ajouter une ombre à un élément, on utilise la propriété *box-shadow*.

Dans le CSS

```
.example {  
    box-shadow: 2px 3px 4px #000;  
}
```

La première valeur (2px) correspond au décalage de l'ombre horizontalement, ici 2px vers la droite. Une valeur négative orienterait l'ombre vers la gauche.

La seconde valeur (3px) correspond au décalage de l'ombre verticalement, ici 3px vers le bas.

La troisième valeur (4px) correspond au flou de l'ombre: zéro correspond à une ombre nette, ici on a un flou de 4px.

La quatrième valeur est la couleur de l'ombre, ici, l'ombre est noire.

Il est possible de faire en sorte que l'ombre soit à l'intérieur de l'élément plutôt qu'à l'extérieur en ajoutant le mot clé "inset" à la fin.

Dans le CSS

```
.example {  
    box-shadow: 2px 3px 4px #000 inset;  
}
```

Il est possible de superposer plusieurs ombres. Il faut séparer les différentes valeurs par des virgules.

Les mêmes règles s'appliquent également pour le texte avec l'utilisation text-shadow:

Dans le CSS

```
.example {  
    text-shadow: 1px 2px 2px #000;  
}
```

Il ne pas hésiter à utiliser des outils en ligne pour générer le code CSS qui peut être parfois complexe. Pour cela recherchez un outil avec les mots clé: **css box shadow generator**